

Amendments to the Claims

1. (Currently Amended) In a computer system, a method of generating an application identifier for distinguishing a software application that is installed on the computer system from among other available software applications for an operating environment of the computer system, the installed software application comprising a plurality of files on the computer system, the files comprising at least an executable file and graphical icon data, the executable file having a name, the method comprising:

obtaining graphical icon data from the files of the installed software application;

applying an identifier generation algorithm to application a combined set of data for the software application, the application combined set of data for the software application comprising the graphical icon data obtained from the files of the installed software application and the name of the executable file of the software application;

generating the application identifier for the software application based on the applying of the identifier generation algorithm, the application identifier operable to distinguish the software application from among the other available software applications on for the operating environment of the computer system; and

utilizing the application identifier for the software application to access information about the software application from a networked database containing information about the software applications available for the operating environment of the computer system; and

display displaying the accessed information about the software application in a graphical user interface;

wherein the identifier generation algorithm is a hashing algorithm.

2. (Cancelled)

3. (Previously Presented) The method of claim 1 wherein the hashing algorithm is a one-way hashing algorithm.

4. (Previously Presented) The method of claim 1 wherein the application identifier is a 20-byte hash value.

5. (Original) The method of claim 1 wherein the graphical icon data is obtained from an application binary.

6. (Original) The method of claim 1 wherein the graphical icon data is obtained from an icon file.

7. (Original) The method of claim 1 further comprising comparing the application identifier with a list of application identifiers to determine an attribute of the software application.

8. (Original) The method of claim 7 wherein the attribute comprises a parental control rating for the software application.

9. (Original) The method of claim 1 further comprising sending the application identifier in a database query.

10. (Original) The method of claim 9 wherein a database receives the database query, and wherein the database returns results indicating whether metadata relating to the software application can be obtained from a metadata service.

11. (Original) The method of claim 9 wherein a database receives the database query, and wherein the database returns results indicating whether the software application is of a particular application type.

12. (Original) The method of claim 1 wherein the application data further comprises a name of the software application.

13. (Original) The method of claim 12 wherein the name is a name of an executable file.

14. (Original) The method of claim 1 wherein the application identifier is a unique fixed-length string.

15. (Original) The method of claim 1 further comprising storing the application identifier in a data file along with one or more other application identifiers for other software applications.

16. (Previously Presented) The method of claim 1 wherein the applying of the identifier generation algorithm comprises using functions included in an application programming interface.

17. (Original) The method of claim 1 wherein the application data further comprises registry data.

18. (Original) The method of claim 1 wherein the software application is a gaming-related software application.

19. (Cancelled)

20. (Currently Amended) In a computer system, a method of generating an application identifier for distinguishing a computer gaming-related software application that is installed on the computer system from among other available software applications for an operating environment of the computer system, the installed software application comprising a plurality of files on the computer system, the files comprising at least an executable file and graphical icon data, the executable file having a name, the method comprising:

applying a hashing algorithm to distinct application binary data for the gaming-related software application, the distinct application binary data at least comprising icon data associated with the software application and a the name for an of the executable file for the software application;

generating a hash value based on the applying of the hashing algorithm to the distinct application binary data, where the hash value is a unique game identifier for the gaming-related

software application, the unique game identifier for use in displaying information associated with the gaming-related software application in a graphical user interface-based gaming activity center;

querying a games metadata service, using the unique game identifier for the software application, to obtain games metadata related to the gaming-related software application; and

displaying information associated with the gaming-related software application in the graphical user interface-based gaming activity center;

wherein the graphical user interface-based gaming activity center displays at least one game that was introduced to the activity center through an automatic search and at least one game that was introduced to the activity center through a manual search.

21. (Original) The method of claim 20 wherein the hashing algorithm is a one-way hashing algorithm.

22. (Original) The method of claim 20 wherein the graphical user interface-based gaming activity center is a feature of an operating system.

23. (Currently Amended) In a computer system, a method of querying a database for information pertaining to a software application installed on the computer system, the software application comprising a plurality of files on the computer system, the files comprising at least an executable file and graphical icon data, the executable file having a name, the method comprising:

obtaining an application identifier for a software application, where the application identifier is a hash value generated by a hashing algorithm, the hash value based on distinct application binary data comprising a combination of at least the graphical icon data associated with the software application and the name of the executable file for the software application;

sending a query to the database, the query comprising information of the application identifier information;

receiving a response to the query from the database; and

utilizing the application identifier for the software application to display information from the response in a graphical user interface.

24. (Original) The method of claim 23 wherein the database comprises a list of gaming-related software applications, and wherein the response to the query comprises an indicator of whether the software application is a gaming-related software application.

25. (Previously Presented) The method of claim 23 wherein the query to the database comprises a request for metadata relating to the software application, and wherein the response to the query comprises metadata relating to the software application.

26. (Previously Presented) The method of claim 23 wherein the query to the database comprises a request for parental control information relating to the software application, and wherein the response to the query comprises parental control information relating to the software application.

27. (Currently Amended) A computer system having a capability to distinguish a software application that is installed on the computer system from among other available software applications for an operating environment of the computer system, the installed software application comprising a plurality of files on the computer system, the files comprising at least an executable file and graphical icon data, the executable file having a name, the computer system comprising:

means for obtaining icon data from the files of a the installed software application;
means for creating an application fingerprint for the software application;

wherein the creating is based on a combined set of data for the software application, the combined set of data comprising at least the icon data obtained from the files of the software application and the name of the executable file of the software application, and the creating comprises applying a hashing algorithm to the icon combined set of data obtained from the software application; and

means for accessing information about the software application utilizing the application fingerprint from a networked database containing information about the software applications available for the operating environment of the computer system; and

means for displaying the accessed information about the software application using the application fingerprint to identify the software application.